



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **08221297 A**(43) Date of publication of application: **30.08.96**

(51) Int. Cl.

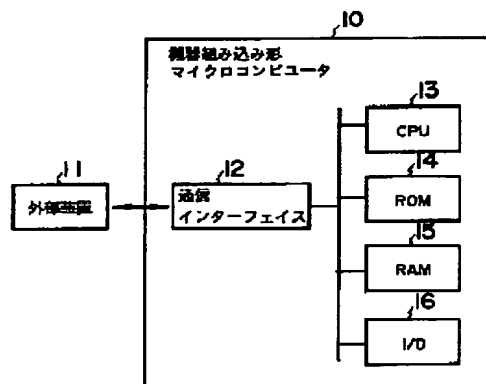
G06F 11/28**G06F 11/22**(21) Application number: **07053617**(22) Date of filing: **16.02.95**(71) Applicant: **CHIYOUFU SEISAKUSHO:KK**(72) Inventor:
YOSHIDA TAKAO
TAKAHAMA HIROYUKI
AZUMA IWAO(54) **PROGRAM DEBUGGING DEVICE**

(57) Abstract:

PURPOSE: To provide an inexpensive program debugging device in which memory data is easy to understand, and a program need not be stopped during the display of the memory data, and the display of the memory data of plural machine built-in type microcomputers is facilitated.

CONSTITUTION: The microcomputer 10 is provided with a CPU 13, a ROM 14 in which an application program and a debugging program to be executed by this CPU 13 are stored, a communication interface 12, and a RAM 15. The CPU 13 executes the application program in response to a debugging command through the communication interface 12 from an external device 11, and executes debugging, and informs the external device 11 of this result, and also, rewrites the constant of the RAM 15 by the command of the external device 11. The external device 11 receives the memory data of the RAM 15 through the communication interface 12, and displays this memory data after converting it by the use of a conversion table.

COPYRIGHT: (C)1996,JPO



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-221297

(43) 公開日 平成8年(1996)8月30日

(51) Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 11/28		7313-5B	G 0 6 F 11/28	L
		7313-5B		P
11/22	3 4 0		11/22	3 4 0 B

審査請求 未請求 請求項の数 3 F D (全 10 頁)

(21) 出願番号 特願平7-53617

(22) 出願日 平成7年(1995)2月16日

(71) 出願人 390002886

株式会社長府製作所

山口県下関市長府扇町2番1号

(72) 発明者 吉田 孝夫

山口県下関市長府扇町2番1号 株式会社
長府製作所内

(72) 発明者 高濱 浩之

山口県下関市長府扇町2番1号 株式会社
長府製作所内

(72) 発明者 東 岩男

山口県下関市長府扇町2番1号 株式会社
長府製作所内

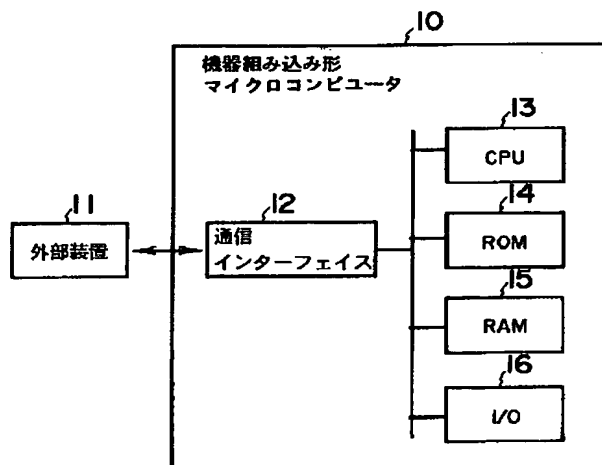
(74) 代理人 弁理士 綾田 正道 (外1名)

(54) 【発明の名称】 プログラムデバッグ装置

(57) 【要約】

【目的】 安価で、メモリデータの理解が容易で、メモリデータ表示中にプログラムを停止する必要がなく、複数の機器組み込み形マイクロコンピュータのメモリデータの表示を容易にした技術の提供。

【構成】 マイクロコンピュータ10に、CPU13と、このCPU13により実行されるアプリケーションプログラムとデバッグ用プログラムとを格納するROM14と、通信インタフェース12と、RAM15とを有する。CPU13は、通信インタフェース12を介する外部装置11からのデバッグ用コマンドに応じアプリケーションプログラムを実行させ、デバッグを行ない、その結果を外部装置11に通知すると共に外部装置11のコマンドによりRAM15の定数の書換えを行なう。外部装置11は、通信インタフェース12を介してRAM15のメモリデータを受信し、このメモリデータを変換テーブルで変換して表示する。



【特許請求の範囲】

【請求項 1】 各種機器に組み込まれる機器組込み形マイクロコンピュータと、前記機器組込み形マイクロコンピュータと接続される外部装置とから成るプログラムデバッグ装置において、前記機器組込み形マイクロコンピュータは、CPUと、このCPUにより実行されるアプリケーションプログラムとこのアプリケーションプログラムをデバッグするためのデバッグ用プログラムとを格納するROMと、前記外部装置と双方向のデータ通信を行う通信インタフェースと、ランダムアクセスおよびデータ書換えが可能なRAMとを有し、前記CPUは、前記通信インタフェースを介して前記外部装置から受信されるデバッグ用コマンドに応じて前記アプリケーションプログラムとデバッグ用プログラムとを実行させ、前記アプリケーションプログラムのデバッグを行ない、前記アプリケーションプログラムデバッグの結果を前記通信インタフェースを介して前記外部装置に通知すると共に、前記通信インタフェースを介して受信する前記外部装置からのコマンドにより前記RAMに格納されている任意の定数の書換えを行ない、前記外部装置は、前記通信インタフェースを介して前記RAMのメモリデータを受信し、前記受信したメモリデータを予め設定した変換テーブルにより変換して表示することを特徴とするプログラムデバッグ装置。

【請求項 2】 各種機器に組み込まれる機器組込み形マイクロコンピュータと、前記機器組込み形マイクロコンピュータと接続される外部装置とから成るプログラムデバッグ装置において、前記機器組込み形マイクロコンピュータは、CPUと、このCPUにより実行されるアプリケーションプログラムとこのアプリケーションプログラムをデバッグするためのデバッグ用プログラムとを格納するROMと、前記外部装置と双方向のデータ通信を行う通信インタフェースと、ランダムアクセスおよびデータ書換えが可能なRAMとを有し、前記CPUは、前記通信インタフェースを介して前記外部装置から受信されるデバッグ用コマンドに応じて前記アプリケーションプログラムとデバッグ用プログラムとを実行させ、前記アプリケーションプログラムのデバッグを行ない、前記アプリケーションプログラムデバッグの結果を前記通信インタフェースを介して前記外部装置に通知すると共に、前記通信インタフェースを介して受信する前記外部装置からのコマンドにより前記RAMに格納されている任意の定数の書換えを行ない、前記外部装置は、前記通信インタフェースを介して前記RAMの複数のメモリデータを受信し、前記受信した複数のメモリデータの各々を予め設定した複数の変換テーブルにより変換して表示することを特徴とするプログラムデバッグ装置。

【請求項 3】 前記機器組込みマイクロコンピュータは複数台から成り、前記複数台の機器組込みマイクロコンピュータの各々の通信インタフェースを切り換えて接続

する切換え部を備え、前記外部装置は、前記複数台の機器組込みマイクロコンピュータの各々のRAMのメモリデータを同時に表示する表示制御部を有することを特徴とする請求項 2 記載のプログラムデバッグ装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、ROM（読出し専用メモリ）に書き込まれたプログラムのデバッグを行なうプログラムデバッグ装置に関する。

10 【0002】

【従来の技術】近年、マイクロコンピュータは、プログラムによってさまざまな用途に適用できる汎用の電子部品として、工作機器や工場の工程制御用装置への組み込み、さらには家電製品への組み込みなど、多岐の分野に適用されている。このような機器組込み形マイクロコンピュータは一般に、ROMにアプリケーションプログラムを格納し、そのプログラムを実行することにより被制御機器と共に多様な機能を実現している。このような機器組込み形マイクロコンピュータのプログラム開発は一般に、従来のプログラムデバッグ装置としてのインサーキットエミュレータ（I E C、In Circuit E m u l a t o r）を用いて行なわれていた。

20 【0003】

【発明が解決しようとする課題】しかしながら、上述したようなインサーキットエミュレータを用いたプログラム開発におけるプログラムデバッグは、インサーキットエミュレータが高価であり、また、機器組込み形マイクロコンピュータに搭載されるCPU（中央処理装置）の種類に応じた専用のインサーキットエミュレータを使用しなければならないため、開発コストが上昇するという問題点があった。また、インサーキットエミュレータはマイクロコンピュータのメモリのデータを10進法、2進法、16進法による数字で表示するのみであり、メモリデータの理解が容易でなかった。さらに、インサーキットエミュレータは機器組込み形マイクロコンピュータのプログラムを停止させなければそのメモリデータを読み出すことができず、インサーキットエミュレータにメモリデータを表示するときには機器組込み形マイクロコンピュータのプログラムの実行を停止する必要があった。さらに、複数の機器組込み形マイクロコンピュータのメモリデータを表示するには複数台のインサーキットエミュレータが必要であった。

【0004】本発明は上記事情を考慮してなされたものであり、その目的とするところは、安価で、メモリデータの理解が容易で、メモリデータ表示中にプログラムを停止する必要がなく、複数の機器組込み形マイクロコンピュータのメモリデータの表示が容易なプログラムデバッグ装置を提供することにある。

【0005】

50 【課題を解決するための手段】この目的を達成するため

に本発明の請求項 1 記載のプログラムデバッグ装置は、各種機器に組み込まれる機器組込み形マイクロコンピュータと、機器組込み形マイクロコンピュータと接続される外部装置とから成るプログラムデバッグ装置において、機器組込み形マイクロコンピュータは、CPUと、このCPUにより実行されるアプリケーションプログラムとこのアプリケーションプログラムをデバッグするためのデバッグ用プログラムとを格納するROMと、外部装置と双方向のデータ通信を行う通信インタフェースと、ランダムアクセスおよびデータ書換えが可能なRAMとを有し、CPUは、通信インタフェースを介して外部装置から受信されるデバッグ用コマンドに応じてアプリケーションプログラムとデバッグ用プログラムとを実行させ、アプリケーションプログラムのデバッグを行ない、アプリケーションプログラムデバッグの結果を通信インタフェースを介して外部装置に通知すると共に、通信インタフェースを介して受信する外部装置からのコマンドによりRAMに格納されている任意の定数の書換えを行ない、外部装置は、通信インタフェースを介してRAMのメモリデータを受信し、受信したメモリデータを予め設定した変換テーブルにより変換して表示する。

【0006】請求項 2 記載のプログラムデバッグ装置は、各種機器に組み込まれる機器組込み形マイクロコンピュータと、機器組込み形マイクロコンピュータと接続される外部装置とから成るプログラムデバッグ装置において、機器組込み形マイクロコンピュータは、CPUと、このCPUにより実行されるアプリケーションプログラムとこのアプリケーションプログラムをデバッグするためのデバッグ用プログラムとを格納するROMと、外部装置と双方向のデータ通信を行う通信インタフェースと、ランダムアクセスおよびデータ書換えが可能なRAMとを有し、CPUは、通信インタフェースを介して外部装置から受信されるデバッグ用コマンドに応じてアプリケーションプログラムとデバッグ用プログラムとを実行させ、アプリケーションプログラムのデバッグを行ない、アプリケーションプログラムデバッグの結果を通信インタフェースを介して外部装置に通知すると共に、通信インタフェースを介して受信する外部装置からのコマンドによりRAMに格納されている任意の定数の書換えを行ない、外部装置は、通信インタフェースを介してRAMの複数のメモリデータを受信し、受信した複数のメモリデータの各々を予め設定した複数の変換テーブルにより変換して表示する。

【0007】請求項 3 記載のプログラムデバッグ装置は、請求項 2 記載のプログラムデバッグ装置において、機器組込みマイクロコンピュータが複数台から成り、複数台の機器組込みマイクロコンピュータの各々の通信インタフェースを切り換えて接続する切換え部を備え、外部装置は、複数台の機器組込みマイクロコンピュータの各々のRAMのメモリデータを同時に表示する表示制御

部を有する。

【0008】

【作用】この構成によって、高価なインサーキットエミュレータを用いることなく一般的なパーソナルコンピュータを用いてアプリケーションプログラムのデバッグを行なうことができるので、安価な構成とすることができ、また、メモリデータが伝送される外部装置の変換テーブルによりメモリデータが変換されて表示されるので、メモリデータの理解が容易となり、さらに、外部装置による機器組込み形マイクロコンピュータのメモリデータの読み出しは極めて短時間に行なわれるので、メモリデータの外部装置での表示中に機器組込み形マイクロコンピュータのプログラムを停止する必要がない。さらに、切換え部により複数の機器組込み形マイクロコンピュータを切り換えることができるので、複数の機器組込み形マイクロコンピュータのメモリデータであっても表示が容易となる。

【0009】

【実施例】以下、本発明の一実施例について図を用いて説明する。図 1 は本発明の一実施例に係るプログラムデバッグ装置を示すブロック図である。図 1 において、10 は各種機器に組み込まれる機器組込み形マイクロコンピュータ、11 は機器組込み形マイクロコンピュータ 10 と後述の通信インタフェースを介して接続される外部装置、12 は RS 232C、RS 422、同期シリアル通信などの一般的な通信インタフェース、13 は CPU、14 は CPU 13 により実行されるアプリケーションプログラムとこのアプリケーションプログラムをデバッグするためのデバッグ用プログラムと外部装置 11 と通信を行なうための通信用プログラムとを格納する ROM、15 はランダムアクセスおよびデータ書換えが可能な RAM、16 は入出力機器との接続を行なうための I/O 部である。

【0010】次に、このような構成のプログラムデバッグ装置の機能、動作について説明する。まず、プログラムデバッグ装置の主機能について説明する。機器組込み形マイクロコンピュータ 10 は、通信インタフェース 12 を介して、ROM 14 に記憶されたアプリケーションプログラムにより外部装置 11 を制御する。外部装置 11 は、機器組込み形マイクロコンピュータ 10 と通信インタフェース 12 を介してデータ通信を行ない、機器組込み形マイクロコンピュータ 10 の RAM 15 のメモリデータを読み出したり、そのメモリ (RAM) 15 に書き込んだりする。また外部装置 11 は、機器組込み形マイクロコンピュータ 10 の実行を制御し、機器組込み形マイクロコンピュータ 10 のメモリデータを変換テーブルによりデータ変換し、変換されたデータを表示する。

【0011】次に、プログラムデバッグ装置の動作について説明する。機器組込み形マイクロコンピュータ 10 のアプリケーションプログラムを起動させた後に外部装

置 1 1 のプログラムを起動する。外部装置 1 1 は、そのプログラムにより一定時間間隔で、機器組込み形マイクロコンピュータ 1 0 に対して、データ通信と停止の指示、停止解除の指示、受信したメモリデータの変換とその表示を繰り返し行なう。機器組込み形マイクロコンピュータ 1 0 は、アプリケーションプログラムを実行しながら、割り込み処理などによって外部装置 1 1 とデータ通信を行なうので、外部装置 1 1 には随時、マイクロコンピュータのメモリデータを表示することができる。そして、そのメモリデータの表示は数字でなく、デバッグする人にわかりやすい表現に変換されているので、仕様通りの動作をしているか否かの判別が容易で、デバッグがしやすい。

【0012】このように本実施例では、高価なインサートキットエミュレータを用いることなく、外部装置 1 1 における結果に基づき、機器組込み形マイクロコンピュータ 1 0 のアプリケーションプログラムのデバッグを行なうことができる。外部装置 1 1 は例えば一般的なパーソナルコンピュータである。また、機器組込み形マイクロコンピュータ 1 0 は ROM 1 4、RAM 1 5、I/O 部 1 6 を内蔵したワンチップマイクロコンピュータでもよい。

【0013】図 2 は図 1 のプログラムデバッグ装置をさらに詳細に示すブロック図である。図 2 において、1 0 は機器組込み形マイクロコンピュータ、1 1 は外部装置、1 3 は CPU、1 4 は ROM、1 5 は RAM、1 6 は I/O 部であり、これらは図 1 と同様なものなので、同一符号を付して説明は省略する。また、1 7 は複数の入出力機器（図示せず）および外部装置 1 1 とデータ通信を行なうための通信インタフェースとしてのシリアル I/O 部、1 8 は複数のシリアル I/O 部を切り換えて接続するシリアル信号切換え部（以下、単に「切換え部」という）、2 0 は機器組込み形マイクロコンピュータ 1 0 と同一構成の機器組込み形マイクロコンピュータ、2 7 は機器組込み形マイクロコンピュータ 2 0 のシリアル I/O 部、1 1 1 はフロッピーディスクドライブ、1 1 2 は CRT、1 1 3 はキーボード、1 1 4 は CPU、1 1 5 はフロッピーディスクドライブ 1 1 1 と接続されるディスクコントローラ、1 1 6 は CRT 1 1 2 と接続されるディスプレイ制御部、1 1 7 はキーボード 1 1 3 と接続される I/O 部、1 1 8 は RAM、1 1 9 は ROM、1 2 0 は機器組込み形マイクロコンピュータ 1 0、2 0 とデータ通信を行なうためのシリアル I/O 部である。

【0014】図 2 において、外部装置 1 1 は一般的なパーソナルコンピュータであり、一般的なパーソナルコンピュータ 1 1 はデータ通信するためのハードウェアを内蔵している。また、機器組込み形マイクロコンピュータ 1 0、2 0 のシリアル I/O 部 1 7、2 7 はそれぞれ切換え部 1 8 と接続されている。さらに、ROM 1 4 には

アプリケーションプログラムと共にデバッグ用プログラムが格納されている。さらに、フロッピーディスクドライブ 1 1 1 には、外部装置 1 1 を制御するプログラムファイルと共に、デバッグのための RAM アドレス設定ファイル、複数の RAM データ変換ファイルが格納されている。

【0015】図 3 はシリアル I/O 部 1 7、2 7 を示すブロック図である。図 3 において、2 0 1 は CPU データバス、2 0 2 はシフトレジスタ、2 0 3、2 0 4、2 0 5 は信号線、2 0 6 はカウンタ、2 0 7 は信号線である。前記図 3 の回路は、組込み用ワンチップマイクロコンピュータなどに多く利用される外部クロック同期シリアル通信回路である。CPU 1 3 は、プログラムにより、CPU データバス 2 0 1 からシフトレジスタ 2 0 2 にデータを書き込み、信号線 2 0 5 に外部クロック XCLK を入力することによりシフトレジスタ 2 0 2 をシフトさせて、シリアル信号 SO を信号線 2 0 4 へ出力する。入力信号 SI のシフトレジスタ 2 0 2 への入力は信号線 2 0 3 を介して行なわれ、外部クロック XCLK の入力によりシフトレジスタ 2 0 2 にシフト入力されると共に、外部クロック XCLK はカウンタ 2 0 6 によりカウントされ、シフトレジスタ 2 0 2 のビット幅のカウントを過ぎると、信号線 2 0 7 を介して CPU 1 3、1 1 4 に対して割り込み要求を発生させる。図 3 の回路では、1 バイト受信する毎に 1 バイト送信される仕組みで送受信を同時に行なう。また、図 3 のシリアル I/O 部は外部クロック XCLK が入力されない限り動作しないし、割り込みが発生しないようになっている。

【0016】図 4 は機器組込み形マイクロコンピュータが 1 台の場合のシリアル I/O 部 1 7、2 7 による外部装置 1 1 との接続例を示す接続図である。図 4 で、左の信号 CLK、SO、SI は外部装置 1 1 における信号、右の信号 XCLK、SIA、SOA は機器組込み形マイクロコンピュータ 1 0 における信号である。外部装置 1 1 から出力される同期通信用クロック CLK は信号線 2 1 1 を介して機器組込み形マイクロコンピュータ 1 0 にクロック XCLK として入力され、外部装置 1 1 から出力されるシリアル信号 SO は信号線 2 1 2 を介して機器組込み形マイクロコンピュータ 1 0 にシリアル信号 SIA として入力され、機器組込み形マイクロコンピュータ 1 0 から出力されるシリアル信号 SOA は信号線 2 1 3 を介して外部装置 1 1 にシリアル信号 SI として入力される。

【0017】図 5 は機器組込み形マイクロコンピュータが 2 台の場合の切換え部 1 8 を示す回路図である。図 5 において、2 2 0 ~ 2 2 9 は信号線、2 3 0 は論理和回路、2 3 1、2 3 2 は論理積回路である。信号線 2 2 0 を介する外部装置 1 1 からの選択信号 SEL により、外部装置 1 1 から信号線 2 2 1 を介して入力されるクロック CLK は 2 台の機器組込み形マイクロコンピュータ 1

0、20に信号線227、224を介してクロックXCLKA、XCLKBとして入力される。信号線222を介して入力される機器組込み形マイクロコンピュータ10、20へのシリアル信号SOは信号線228、信号線225を介してシリアル信号SIA、SIBとして機器組込み形マイクロコンピュータ10、20に入力される。2台の機器組込み形マイクロコンピュータ10、20からのシリアル信号SOA、SOBは論理和回路230により論理和され、信号線223を介するシリアル信号SIとなって外部装置11に入力される。(表1)に、フロッピーディスクドライブ111に格納されているRAMアドレス設定ファイルの一例を示す。

【0018】

【表1】

label101	chip1	b0h	DEC	r
label202	chip1	b1h	HEX	r
label303	chip1	b2h	DEC	r
label401	chip1	b3h	HEX	w
label15	chip1	b4h	DEC	w
label16	chip1	b5h	HEX	w
label17	chip2	b6h	DEC	w
label18	chip2	b7h	HEX	w
label110	chip1	b8h	DEC	w
label111	chip1	bah	DEC	w
label112	chip2	bbh	HEX	w

【0019】(表1)に示すファイルの内容は文字列データが1変数1行に記述されている。行頭から、プログラムの変数(label11、label12、……)、変換の必要があれば記号@に続く変換ファイル名(@1、@2、@3)、CPUチップ番号(chip1、chip2)、16進RAMアドレス(b0h、b1h、……)、表示形式(DEC、HEX)、書込み許可指示(r、w)の順で記述する。表示形式ではDECは10進、HEXは16進を表し、書込み許可指示ではrは読出しのみ、wは書込み許可を表す。(表2)に、フロッピーディスクドライブ111に格納されているRAMデータ変換ファイルの一例を示す。

【0020】

【表2】

ファイル1の内容

1	状態A
2	状態B
3	状態C
4	状態D
5	状態その他

ファイル2の内容

1	ステップA
2	ステップB
3	ステップC
4	ステップD
5	ステップその他

ファイル3の内容

1	データA
2	データB
3	データC
4	データD
5	データその他

* (表2)に示すそれぞれのファイルは(表1)のRAMアドレス設定ファイルに指定されている変換ファイルである。ファイルには1行に1つの値とその値が表す任意の文字列とが記述されている。RAMデータが取り得る全ての値を記述する必要はない。

【0021】図6は、機器組込み形マイクロコンピュータ10、20のアプリケーションプログラムの内容を示すフローチャートである。図6において、まずシリアルI/O部17、27の初期化を行ない(ステップS1)、停止フラグがオンか否かを判別する(ステップS2)。停止フラグがオンと判別した場合は、なにも処理せずループし、停止フラグがオンでないと判別した場合は、機器本来の処理を行ない続ける(ステップS3)。

【0022】図7は、機器組込み形マイクロコンピュータ10、20における外部装置11からの割り込みルーチン、すなわちデバッグ用プログラム内容のルーチンを示すフローチャートである。図6のプログラムの実行中に外部装置11からのシリアル通信(シリアル信号入力)があると、図3のシリアルI/O部17、27より割り込みが発生する。この割り込みにより実行されるルーチンでは、コマンドの種類によって、RAM読出し処理(ステップS11、S12)、RAM書込み処理(ステップS13、S14)、停止処理(ステップS15、S16)、停止解除処理(ステップS17、S18)を行なう。割り込みルーチンはデバッグ用ルーチンであり、機器本来の処理とは無関係であるが、外部装置11からのシリアル信号入力がない限り実行されないの、最終的に製品のプログラムに残しても問題はない。

【0023】図8は、上記シリアル通信における通信手順を説明するためのデータ図である。このシリアル通信では図8に示すように3バイトを基本にしている、1バイト目はコマンド、2バイト目はアドレス、3バイト目はデータを送受信するようにプログラムされている。1バイト目のコマンドは、1が停止解除、2が停止、3が読出し、4が書き込みのコマンドである。2バイト目のアドレスは0から255まで表すことができ、3バイト目のデータも0から255まで表すことができる。上記3バイトは、外部装置11と機器組込み形マイクロコンピュータ10、20とが図3のシリアルI/O部17、27により同時に送受信を行なうことにより通信される。しかし、コマンドバイトは外部装置11から出力され、停止解除、停止、読出し、書き込みに対する数値は上述のように定められている。2バイト目のアドレスバイトは外部装置11から出力され、機器組込み形マイクロコンピュータ10、20のRAMアドレスを指定する。3バイト目のデータバイトは外部装置11から機器組込み形マイクロコンピュータ10、20のRAMにデータを書き込む場合は有効であるが、読出しの場合は3バイト目のデータは廃棄され、機器組込み形マイクロコンピュータ10、20の出力データのみが有効となる。

【0024】図9は、外部装置11のプログラムを説明するためのフローチャートである。上記プログラムでは、はじめに表1のRAMアドレス指定ファイルとRAMデータ変換ファイルをフロッピーディスクドライブ111(図2)から読み出し、RAM118にセットする(ステップS31)。RAM118にセットされた変換テーブル文字列配列データは変換テーブルを構成する。BASIC言語でRAMアドレス設定配列データと変換テーブル文字列配列データを記述すると、それぞれ次のようになる。

DIM RAMDEF(10)(5)

DIM TRANSTBL\$(10)(256)

ここで、RAMアドレス設定配列データは10個のRAMデータを扱い、変数は次の意味を表す。

RAMDEF(I)(1)はI番目のRAMアドレス変数のRAMアドレス

RAMDEF(I)(2)はI番目のRAMアドレス変数の変換テーブル番号

RAMDEF(I)(3)はI番目のRAMアドレス変数の表示データ形式

RAMDEF(I)(4)はI番目のRAMアドレス変数のCPU選択番号

RAMDEF(I)(5)はI番目のRAMアドレス変数の書き込み許可フラグ

また、文字列変数TRANSTBL\$(I)(J)はI番目の変換テーブルの値Jに相当する文字列である。

【0025】次に、キー入力コマンドがあるか否かを判別する(ステップS32)。キー入力なしと判別した場

合には、RAMアドレス設定配列データに指定されたアドレスの機器組込み形マイクロコンピュータ10、20のRAM15のデータを読み出し(ステップS33)、RAMアドレス設定配列データに指定されている変換テーブル番号の変換テーブル文字列配列データの中から、RAMデータの値に対応する文字列を検索し(ステップS34)、これを表示する(ステップS35)。変換テーブル番号が指定されていない場合には表示データ形式にて表示する。また、指定した番号の変換テーブルに、読み出したRAMデータの値に等しい文字列がない場合には表示しない(ステップS35)。この一連の処理を繰り返すことにより、実行中の機器組込み形マイクロコンピュータ10、20の指定したRAMデータの内容が数値及び変換された文字列でCRT112に表示され続ける。この表示内容の一例を(表3)に示す。

【0026】

【表3】

label101	chip1	b0h	DEC r	RAM=1	状態A
	.			.	
	.			.	

【0027】(表3)におけるlabel~DEC rは前述した通りの内容である。同一行の次のRAM=1はRAMの「b0」番地データが「1」であることを示し、状態Aは(表2)に示すようにRAM=1に対応する文字列である。状態Aとは例えばサーミスタ温度 $t=60^{\circ}\text{C}$ のような状態である。ステップS32でキー入力コマンドがあると判別された場合には、そのキー入力が見すコマンドキャラクタにより、RAM書き込み処理(ステップS36、S37)、停止処理(ステップS38、S39)、停止解除処理(ステップS40、S41)の各コマンド処理を行なう。

【0028】次に、このようなハードウェア、ソフトウェア構成のプログラムデバッグ装置の動作について説明する。最初に機器組込み形マイクロコンピュータ10、20が起動されると、この時点で停止フラグはオフに初期化されているので(図6のステップS1)、機器本来の処理を行なう(ステップS2、S3)。その後、外部装置11およびそのプログラムが起動されると、図9のプログラムにより、機器組込み形マイクロコンピュータ10、20のRAMデータを数値および分かりやすく変換された文字列でCRT112によりモニタすることができる。これで、機器本来の仕様通りに動作しているか否かをRAM15格納の内部変数(メモリデータ)で確認しながら調べることができる。機器に異常があった場合、停止コマンドをキーボード113から入力して、機

器組込み形マイクロコンピュータ10、20の動作を停止させて（ステップS38、S39）、上記内部変数を確認したり、内部変数を変更したりできる。また、機器組込み形マイクロコンピュータ10、20の動作中に上記内部変数を変更して、最適な定数を設定することもできる（ステップS36、S37）。さらに、機器組込み形マイクロコンピュータ10、20の動作中に、外部装置11のプログラムを停止させてもコンピュータ10、20には全く影響せず、コンピュータ10、20は動作を継続できる。なお、本実施例ではコンピュータ10、20のRAMが1個の場合を示したが、RAMが複数であっても、外部装置11が、通信インタフェース17を介してRAMの複数のメモリデータを受信し、受信した複数のメモリデータの各々を予め設定した複数の変換テーブルにより変換して表示するようにすれば、上記複数のメモリに対して対応が可能となる。

【0029】このように、本実施例では、機器組込み形マイクロコンピュータ10、20を切換え部18を介して外部装置11、すなわち図2では一般的なパーソナルコンピュータ11と接続するようにしたので、外部装置11における表示内容に基づいて、機器組込み形マイクロコンピュータ10、20のROM14格納のプログラムのデバッグが可能となる。また、外部装置11のプログラムを変更する必要はなく、機器組込み形マイクロコンピュータ10、20に組み込むデバッグ用プログラムを変更するだけで、多種多様の機器組込み形マイクロコンピュータに対応可能である。さらに、通信インタフェースとしてのシリアルI/O部17、27はインサートキットエミュレータで使用するような高い周波数で動作することはないので、本装置はノイズにも強く、現場でのデバッグ作業にも適している。

【0030】

【発明の効果】以上のように本発明は、機器組込み形マイクロコンピュータに、CPUと、このCPUにより実行されるアプリケーションプログラムとこのアプリケーションプログラムをデバッグするためのデバッグ用プログラムとを格納するROMと、外部装置と双方向のデータ通信を行う通信インタフェースと、ランダムアクセスおよびデータ書換えが可能なRAMとを有し、CPUは、通信インタフェースを介して外部装置から受信されるデバッグ用コマンドに応じてアプリケーションプログラムとデバッグ用プログラムとを実行させ、アプリケーションプログラムのデバッグを行ない、アプリケーションプログラムデバッグの結果を通信インタフェースを介して外部装置に通知すると共に、通信インタフェースを介して受信する外部装置からのコマンドによりRAMに格納されている任意の定数の書換えを行ない、外部装置は、通信インタフェースを介してRAMのメモリデータを受信し、受信したメモリデータを予め設定した変換テーブルにより変換して表示するようにしたことにより、

高価なインサートキットエミュレータを用いることなく一般的なパーソナルコンピュータを外部装置として用いてアプリケーションプログラムのデバッグを行なうことができるので、安価な構成とすることができる。また、メモリデータが伝送される外部装置の変換テーブルによりメモリデータが変換されて表示されるので、メモリデータの理解が容易となり、さらに、外部装置による機器組込み形マイクロコンピュータのメモリデータの読み出しはデバッグ用プログラムにより極めて短時間に行なわれるので、メモリデータの外部装置での表示中に機器組込み形マイクロコンピュータのプログラムを停止する必要がない。

【0031】さらに、外部装置が、通信インタフェースを介してRAMの複数のメモリデータを受信し、受信した複数のメモリデータの各々を予め設定した複数の変換テーブルにより変換して表示することにより、複数のメモリに対して対応が可能となる。

【0032】さらに、切換え部により複数の機器組込み形マイクロコンピュータを切り換えることができるので、複数の機器組込み形マイクロコンピュータに対してあたかも1台の場合と同等の対応が可能であり、上記と同様の効果を奏することができる。

【図面の簡単な説明】

【図1】本発明の一実施例に係るプログラムデバッグ装置を示すブロック図である。

【図2】図1のプログラムデバッグ装置を詳細に示すブロック図である。

【図3】図2のシリアルI/O部を示すブロック図である。

【図4】機器組込み形マイクロコンピュータが1台の場合のシリアルI/O部を示す接続図である。

【図5】図2のシリアル信号切換え部を示す回路図である。

【図6】機器組込み形マイクロコンピュータのアプリケーションプログラムを説明するためのフローチャートである。

【図7】機器組込み形マイクロコンピュータの割り込みルーチンを示すフローチャートである。

【図8】シリアル通信における通信手順を説明するためのデータ図である。

【図9】外部装置のプログラムを説明するためのフローチャートである。

【符号の説明】

10、20 機器組込み形マイクロコンピュータ
11 外部装置
12 通信インタフェース
13、114 CPU
14、119 ROM
15、118 RAM
16、117 I/O部

13

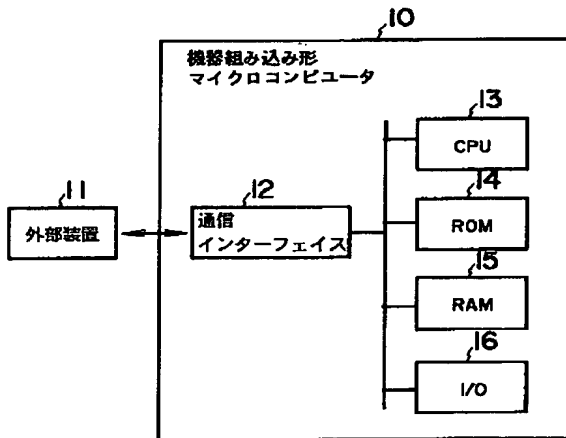
14

17、27、120 シリアルI/O部
 18 シリアル信号切換え部
 111 フロッピーディスクドライブ
 112 CRT
 113 キーボード

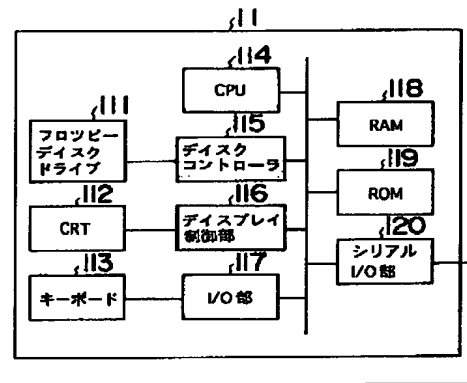
* 113 キーボード
 115 ディスクコントローラ
 116 ディスプレイ制御部

*

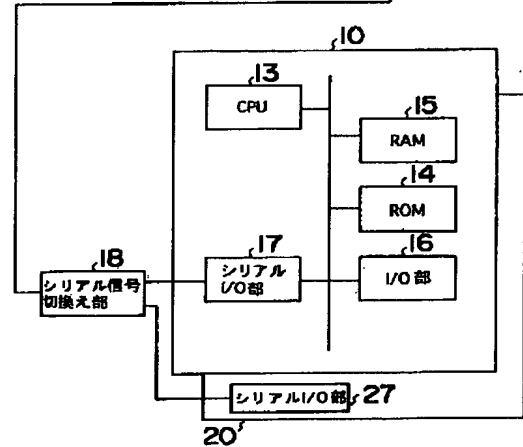
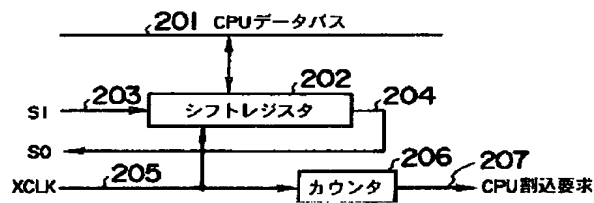
【図1】



【図2】

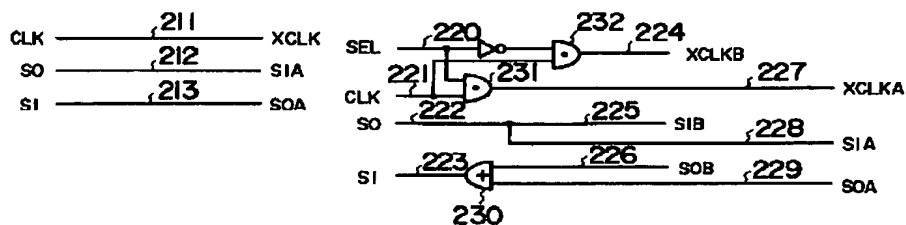


【図3】

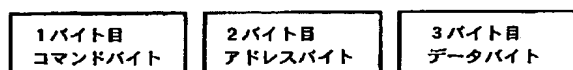


【図4】

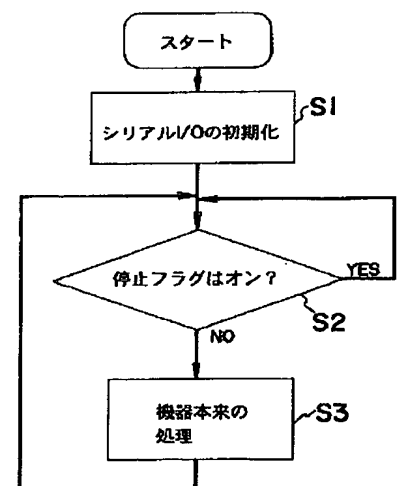
【図5】



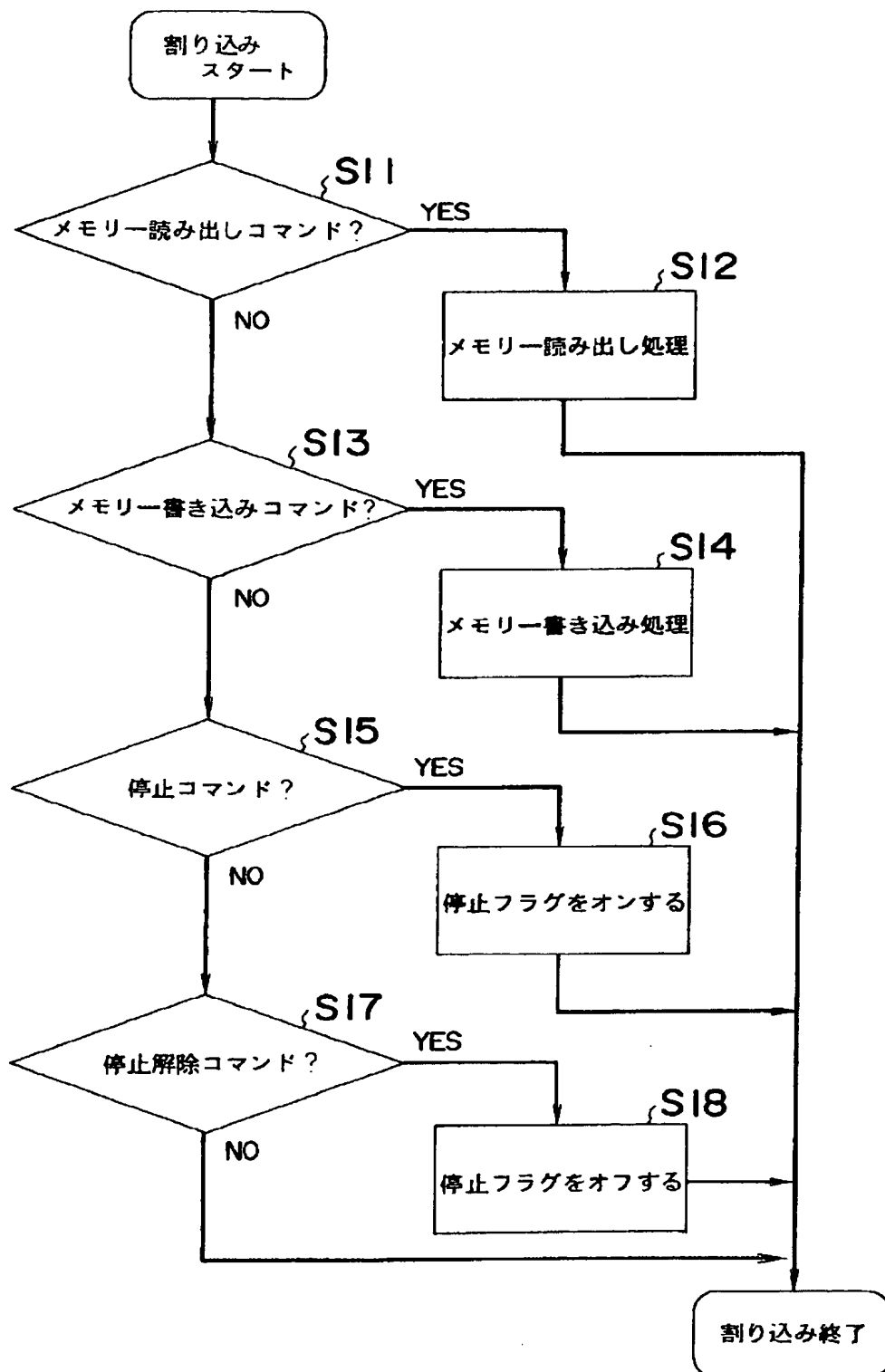
【図8】



【図6】



【図 7】



【図 9】

